

REMARKS

Claims 1-5, 7-12 and 32-36 are pending. Claims 1, 5, 7 and 32 are amended. Claim 6 is cancelled. Claims 13-31 are withdrawn from further consideration, without prejudice or disclaimer to be re-filed at a later date.

The claim amendments are supported by the application as originally filed, for instance, originally filed claims 6 and 7, Figs. 1, 2 and 4, and the specification at page 6, lines 12-20 and page 12, line 14 through page 14, line 15. No new matter has been added.

The Office Action included the following rejections:

- Claims 5 and 32 were rejected under 35 USC § 112;
- Claims 1-9, 12 and 32-36 were rejected under 35 USC § 102(e) as anticipated by Lee, U.S. Patent No. 5,862,377; and
- Claims 10 and 11 were rejected under 35 USC § 103 as obvious in view of Lee and Reisman, U.S. Patent No. 6,594,692.

Applicant respectfully submits that the claims, as newly amended, are patentably distinct from the cited art. Applicant respectfully requests that the above rejections be withdrawn in view of the claim amendments and remarks below.

35 USC § 112

Claim 5 has been amended, as set forth above, to replace the term "database" with "data schema." Antecedent basis is provided for the term, "data schema," in claim 1.

Reconsideration is respectfully requested.

Claim 32 has been amended, as set forth above, to clarify the language of that claim. Reconsideration is respectfully requested.

35 USC § 102(e)

Claim 1, as newly amended, defines a software architecture having at least one feature which Lee fails to disclose or suggest, namely:

a data schema for storing a plurality of data objects, the data schema having an underlying, extensible data model providing a configuration of the data objects in the data schema in terms of fixed attributes and extensible attributes, the extensible attributes of the data model enabling extension of the data schema with addition of a previously undefined attribute without having to alter the configuration of the data model . . .

In some embodiments, the software architecture defined in claim 1 enables multiple non-technical or lay users to perform tasks for building a complete network-based, multiuser application via a wide-area network. For example, the platform can be used to construct and maintain a customer relationship management system. The data schema and underlying, extensible data model enable users of the architecture to extend the data schema in a user-specific way. Users and applications can access and use the same data schema without having to change the underlying data model configuration. For example, data objects can be extended as new relationships form which require enforcement of data dependencies, without having to add new rules to support the new data.

The Office Action states that Lee teaches a data schema for storing a plurality of data objects with an underlying extensible data model at col. 8, lines 45-57. The quoted passage of Lee describes a language or protocol, namely Object Control Language ("OCL") which applications use to communicate information between one another. The quoted passage of Lee states that OCL is "an object oriented data architecture providing an extensible format for exchanging data between applications." The Office Action is apparently attempting to equate OCL with a data schema having an underlying, extensible data model. Lee simply does not support this assertion, as there are no similarities on which to base the comparison.

While Lee characterizes Object Control Language as an “architecture,” OCL is in fact what its name suggests, a language. OCL is neither a data schema nor an underlying extensible data model, per claim 1. OCL is simply a language or protocol for communicating information between two applications. Lee offers no suggestion that OCL includes or operates in conjunction with a “data schema having an underlying, extensible data model providing a configuration of the data objects in the data schema in terms of fixed attributes and extensible attributes, the extensible attributes of the data model enabling extension of the data schema with addition of a previously undefined attribute without having to alter the configuration of the data model,” as recited in claim 1. Instead, Lee teaches a registry with tokens which sender and receiver applications can access to generate interfaces and communicate with one another using OCL. Neither OCL nor the token registry described in Lee have anything to do with a data schema or underlying extensible data model, much less a data model “providing a configuration of the data objects ... in terms of fixed attributes and extensible attributes.” Claim 1, on the other hand, provides a data schema having an underlying, extensible data model that, for example, enables users of the architecture to extend the data schema in a user-specific way without having to change the underlying data model configuration.

Lee also describes an X Window System, and defines it as a “network transparent graphical windowing interface for computers.” (col. 3, lines 57-59). While programmers are able to write applications using interface components in the form of widgets 250 and intrinsics 240, Lee does not suggest that these tools are somehow a data schema or underlying extensible data model, much less a data model “providing a configuration of the data objects ... in terms of fixed attributes and extensible attributes,” as recited in claim 1. Lee only suggests that widgets 250 and intrinsics 240 are general user interface building tools:

“Intrinsics 240 provide a framework for combining the widgets to create a complete user interface.” (col. 4, lines 46-47).

Lee further describes a low level X interface 230 of the X Window System that “provides the application with access to the most fundamental procedures and data structures of the X Window System.” (col. 4, lines 47-50). Again there is no suggestion that this low level interface or related data structures of X Windows provide a “data schema having an underlying, extensible data model providing a configuration of the data objects in the data schema in terms of fixed attributes and extensible attributes, the extensible attributes of the data model enabling extension of the data schema with addition of a previously undefined attribute without having to alter the configuration of the data model,” as recited in claim 1. It would be improper to use hindsight to attempt to read these features of the software architecture of claim 1 into a description of a conventional X Window System.

In short, because Lee’s teaching of the OCL and the X Window System include no mention of a data schema or underlying, extensible data model, a conventional software architecture should be assumed. It is improper to use hindsight to propose that the novel and nonobvious software architecture described and claimed in the present application appears in the Lee reference.

Because Lee fails to disclose or suggest the above features of the software architecture defined in claim 1, Lee fails to support an anticipation rejection of claim 1 under 35 USC § 102(e). This rejection should be withdrawn.

35 USC § 103

Reisman fails to cure the deficiencies of Lee, considered alone or in combination with Lee under 35 USC § 103. Reisman teaches a transport component for distributing information to computer stations in a network. While Reisman describes e-commerce

transactions in this context, Reisman makes no mention of a data schema or underlying extensible data model, much less a data model "providing a configuration of the data objects ... in terms of fixed attributes and extensible attributes." Claim 1, on the other hand, recites this feature and, therefore, is able to provide the benefit of enabling users of the architecture to extend a data schema in a user-specific way without having to change an underlying data model configuration. It would be improper to use hindsight to assert that the novel and nonobvious software architecture defined in claim 1 underlies the Reisman transport component.

Because Reisman fails to disclose or suggest the above-quoted features of claim 1, Reisman fails to support an obviousness rejection of claim 1, taken alone or in combination with Lee. This rejection should be withdrawn.

The Dependent Claims

The remaining claims are dependent upon claim 1 and are, therefore, patentable for at least the same reasons as claim 1. Reconsideration is respectfully requested.

CONCLUSION

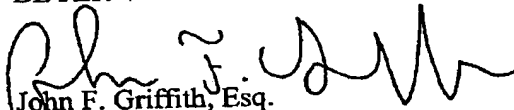
In view of the above Amendments and Remarks, Applicant submits that this application is in condition for allowance. Early notification to that effect is respectfully requested.

Should the Examiner believe that a telephone conference would expedite the prosecution of this application, Applicant's attorney can be reached at the number below.

If any fees are due in connection with the filing of this amendment (including any fees due for an extension of time), such fees may be charged to Deposit Account No. 500388 (Order No. BIG1P001).

Respectfully submitted,

BEYER WEAVER & THOMAS, LLP


John F. Griffith, Esq.
Reg. No. 44,137

P.O. Box 70250
Oakland, CA 94612-0250